



REC'D 26 AUG 2003

WIPO PCT

**Prioritätsbescheinigung über die Einreichung  
einer Patentanmeldung**

BEST AVAILABLE COPY

**Aktenzeichen:** 102 31 970.7

**Anmeldetag:** 15. Juli 2002

**Anmelder/Inhaber:** Siemens Aktiengesellschaft, München/DE

**Bezeichnung:** Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur

**IPC:** G 06 F 17/30

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 17. Juli 2003  
Deutsches Patent- und Markenamt  
Der Präsident  
Im Auftrag

Ebert

**PRIORITY DOCUMENT**  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)



## Beschreibung

Verfahren zur Codierung von Positionen von Datenelementen in  
5 einer Datenstruktur

Die Erfindung betrifft ein Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur.

10 In einer Datenstruktur treten häufig Datenelemente auf, welche durch ihre Positionen zueinander unterschieden werden sollen. Bei Positionscodierverfahren wird dies dadurch ermöglicht, dass für alle Datenelemente in einer vorgegebenen Reihenfolge Positionscodes vergeben werden.

15

In [1] ist ein Positionscodierverfahren beschrieben, welches bei einem Verfahren zur binären Codierung von XML-Daten eingesetzt wird. Dieses Verfahren verwendet XML-Schema-Definitionen (beispielsweise im Rahmen eines standardisierten MPEG-  
20 7 Verfahrens), um die Codes für die einzelnen Datenelemente der XML-Beschreibung zu generieren. Dabei können einzelne Elemente bzw. Elementgruppen des gleichen Typs gemäß der XML-Schema-Definition mehrmals im Dokument auftreten. In diesem Fall wird ein Positionscore (PC) übertragen. Der Positionscore ist die binäre Repräsentation einer ganzen Zahl, welche  
25 die Position bezüglich der Nachbarelemente spezifiziert. Der Positionscore wird dem Element aufgrund der Position zu den Nachbarelementen in dem zu codierenden Dokument zugewiesen. Dies hat den Vorteil, dass der Positionscore eines Elementes  
30 erhalten bleibt, unabhängig von der Reihenfolge, in der benachbarte Elemente übertragen werden. Somit können bei der Übertragung Elemente verloren gehen, ohne dass dies die Positionscodes bzw. die Position der anschließend durch einen Decoder decodierten Elemente beeinflusst.

35

Ein Nachteil dieses bekannten Verfahrens ist, dass zum Zeitpunkt der Codierung das XML-Dokument bekannt sein muss, damit der bisher existierenden Positionscodierung keine neuen

Positionen eingefügt werden können, sondern lediglich neue Positionen angehängt werden können. Dies ist insbesondere dann ein Nachteil, wenn während der Erstellung eines XML-Dokumentes dieses bereits codiert bzw. übertragen werden soll, beispielsweise in Live-Übertragungen nach einem Übertragungs-Standard, z.B. MPEG-4 oder MPEG-7.

Zur Lösung dieses Problems können zwischen den verwendeten Positionscodes Lücken gelassen werden, die bei Bedarf aufgefüllt werden können. Insbesondere bei Live-Encodierung ist jedoch die vorab festzulegende, bedarfsgerechte Bereithaltung solcher Lücken schwer vorhersehbar. Außerdem ist die Gesamtanzahl möglicher Lücken in vielen Fällen durch die XML-Schema-Definition beschränkt. Stehen nun an der einzufügenden Position keine durch solche Lücken freigehaltenen Positionscodes mehr zur Verfügung, müssen alle bereits gesendeten, benachbarten Elemente mit neu generierten Positionscodes erneut übertragen werden. Insbesondere kommt dies häufig bei mehreren Datenelementen des gleichen Typs vor, z.B. bei mehrfach in einem Dokument auftretenden identischen Elementen bzw. Elementgruppen. Die Folge ist eine deutliche Verschlechterung der Codiereffizienz sowie ein deutlich erhöhter Verarbeitungsaufwand sowohl am Encoder als auch am Decoder.

Aufgabe der Erfindung ist es deshalb, ein Verfahren und eine Vorrichtung zur Codierung von Positionen von Datenelementen in einer Datenstruktur zu schaffen, bei denen auf einfache und effiziente Weise die Positionen von neu hinzukommenden Datenelementen codiert werden können.

30

Diese Aufgabe wird durch das Verfahren gemäß Anspruch 1 bzw. 2 und die Vorrichtung gemäß Anspruch 13 bzw. 14 gelöst. Vorteilhafte Ausgestaltungen der Erfindung sind in den Unteransprüchen beschrieben.

35

Das erfindungsgemäße Verfahren hat den Vorteil, dass die Positionscodierung robust gegenüber Datenverlust ist, da Posi-

tionscodes beibehalten werden. Zugleich können bei der Verwendung des Verfahrens zur Codierung von XML-Dokumenten dynamische Dokumente, die während der Codierung erzeugt werden, effizient codiert werden. Dies wird dadurch ermöglicht, dass neue Positionen zwischen bestehenden Positionen codiert werden können, ohne dass Elemente und deren Positionscodes erneut übertragen werden müssen.

Eine Ausführungsform der Erfindung wird nachfolgend anhand der beigefügten Zeichnungen erläutert.

Es zeigen:

Fig. 1 eine Darstellung eines Positionscodes eines Datenelements, wobei der Positionscod mithilfe des erfindungsgemäßen Verfahrens erzeugt wurde;

Fig. 2 eine Datenstruktur, wobei den Datenelementen Positionscodes zugeordnet sind, die mithilfe des erfindungsgemäßen Verfahrens erzeugt wurden;

Fig. 3 die Datenstruktur gemäß Fig.2, wobei zwei neue Datenelemente hinzugefügt worden sind.

In der nachfolgend betrachteten Ausführungsform der Erfindung werden den Datenelementen der Datenstruktur in aufsteigender Reihenfolge der Datenelement-Positionen Positionscodes zugewiesen, die ebenfalls in aufsteigender Reihenfolge angeordnete rationale Zahlen in einem vorgegebenen Wertebereich sind. Ist nun eine Position zwischen zwei existierenden Positionen zu adressieren, so ist dies immer möglich, da zwischen zwei gegebenen rationalen Zahlen  $R_1$  und  $R_2$  mit  $R_1 \neq R_2$  immer eine infinitesimale Anzahl an rationalen Zahlen existiert. In realen Implementierungen ist diese Anzahl zwar nicht infinitesimal, kann aber immer ausreichend groß gewählt werden, beispielsweise  $>1024$ . Wenn der Positionscod des ersten Datenelements ungleich Null ist, können auch Datenelemente eingefügt wer-

den, deren Positionscode kleiner als der Positionscode des ersten Datenelements ist.

Die Verwendung von rationalen Zahlen hat den weiteren Vorteil, dass eine möglichst kurze Binärdarstellung ermöglicht wird.

In Fig. 1 ist ein Positionscode eines Datenelements gezeigt. Dieser Positionscode ist die Binär-Darstellung einer rationalen Zahl zur Basis 2 im Wertebereich  $]0,1[$ . Die Binär-Darstellung der rationalen Zahl umfasst  $N=15$  Bits, wobei  $N^*=12$  Datenbits (MSB-Bit, Bit 1 bis Bit 11;  $N^* \leq N$ ) vorhanden sind, welche in drei Quadrupeln angeordnet sind. Die Wertigkeiten der Datenbits sind jeweils unter den Bits genannt. Den Datenbits sind drei Extension-Bits vorangestellt, wobei die Anzahl der Extension-Bits die Anzahl der vorhandenen Datenbit-Quadrupel angibt. Die zwei vorderen Extension-Bits sind auf eins gesetzt, und das letzte Extension-Bit ist auf Null gesetzt. Durch das Setzen des letzten Extension Bits auf Null wird signalisiert, dass es sich bei den nachfolgenden Bits um Datenbits handelt. Mit der in Fig. 1 gewählten Darstellung wird somit eine rationale Zahl durch  $N$  Bits repräsentiert, von denen  $N^*$  Bits Daten-Bits sind, wobei  $N^* \leq N$  und  $N^*=4k$  ist ( $k$  ist dabei eine ganze Zahl im Wertebereich  $[1, \infty[$ ).

25

In Fig. 2 ist eine Datenstruktur in der Form eines Datenbaums gezeigt, wobei die Positionscodes der Datenelemente mit dem vorstehend beschriebenen Verfahren erzeugt wurden. Die Datenstruktur umfasst ein Datenelement A, das mit fünf Datenelementen B verknüpft ist. Den Datenelementen B sind Positionscode  $P$  in aufsteigender Reihenfolge in der Form von rationalen Zahlen  $1/8, 1/4, 3/8, 1/2$  bzw.  $5/8$  zugeordnet. Ferner sind die Binärdarstellungen der Positionscodes gemäß der Darstellung der Fig. 1 angegeben.

35

In Fig. 3 ist eine Datenstruktur gemäß Fig. 2 gezeigt, wobei zwischen dem Datenelement mit dem Positionscodex 3/8 und dem Datenelement mit dem Positionscodex 1/2 noch zwei neue Datenelemente eingefügt worden sind. Diese neu hinzugefügten Datenelemente sind in Fig. 3 grau dargestellt. Durch die Verwendung von rationalen Zahlen für die Positionscodes können nun zwei Werte für die Positionscodes der neuen Datenelemente gefunden werden, die zwischen den Werten 3/8 und 1/2 liegen. In Fig. 3 wurden für diese Werte die Zahlen 7/16 und 15/32 gewählt. Es ist folglich möglich, neue Positionscodes für neue Datenelemente in der Datenstruktur zu generieren, ohne dass die bestehenden Positionscodes verändert werden müssen. Somit können zugewiesene Positionscodes bestehen bleiben und an beliebiger Position beliebig viele neue Datenelemente eingefügt werden.

Literaturverzeichnis:

- [1] ISO/IEC 15938-1 Multimedia Content Description Interface  
- Part 1: Systems, Geneva 2002

## Patentansprüche

1. Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur, bei dem

5

den Datenelementen Positionscodes in einer vorgegebenen Reihenfolge zugeordnet werden,

dadurch gekennzeichnet, dass

10

die Positionscodes derart gewählt werden, dass im Falle von unbegrenzten Codelängen der Positionscodes zwischen

den Positionen von zwei Datenelementen beliebig viele

weitere Positionscodes zur Codierung von Positionen von

weiteren Datenelementen vergeben werden können.

15

2. Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur, bei dem

den Datenelementen Positionscodes in einer vorgegebenen Reihenfolge zugeordnet werden,

dadurch gekennzeichnet, dass

20

die Positionscodes derart gewählt werden, dass zwischen den Positionen von zwei benachbarten Datenelementen wei-

tere Positionscodes zur Codierung von Positionen von

weiteren Datenelementen vergeben werden können, wobei

die Codelänge von mindestens einem weiteren Positionsc-

ode größer als die längere der Codelängen der Positions-

codes der zwei benachbarten Datenelemente ist.

3. 3. Verfahren nach Anspruch 1 oder 2, dadurch ge-

kennzeichnet, dass die Positionscodes rationale

30

Zahlen repräsentieren.

4. Verfahren nach Anspruch 1 oder 2 oder 3, dadurch

gekennzeichnet, dass der erste und/oder der letz-

te Positionscodes der Datenelemente derart gewählt wer-

35

den, dass vor dem ersten und/oder nach dem letzten Posi-

tioncode weitere Positionscodes eingefügt werden kön-

nen.



5. Verfahren nach Anspruch 4, dadurch gekennzeichnet, dass der erste Positionscodex ungleich null ist und/oder der letzte Positionscodex ungleich eins ist.
- 5 6. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Positionscodes Binärdaten sind.
- 10 7. Verfahren nach Anspruch 6, dadurch gekennzeichnet, dass die Positionscodes ein oder mehrere Datenbit-n-Tupel und ein oder mehrere Extension-Bits umfassen, wobei die Anzahl der Extension-Bits der Anzahl der Datenbit-n-Tupel entspricht.
- 15 8. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Datenstruktur Teil eines Datenbaums ist.
- 20 9. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die Datenelemente Datencodes für die Datenelemente eines Dokuments sind.
- 25 10. Verfahren nach Anspruch 9, dadurch gekennzeichnet, dass das Dokument ein XML-Dokument ist.
11. Verfahren nach Anspruch 9 oder 10, dadurch gekennzeichnet, dass die Datencodes für das Dokument mit einem MPEG-Codierverfahren erzeugt werden.
- 30 12. Verfahren nach Anspruch 11, dadurch gekennzeichnet, dass das Codierverfahren ein standardisiertes MPEG-7 Codierverfahren ist.
- 
- 35 13. Vorrichtung zur Codierung von Positionen von Datenelementen in einer Datenstruktur, dadurch gekennzeichnet, dass mit der Vorrichtung ein Verfahren

nach einem der vorhergehenden Ansprüche durchführbar ist.

- 5 14. Vorrichtung zur Decodierung von Positionscodes von Datenelementen in einer Datenstruktur, dadurch gekennzeichnet, dass mit der Vorrichtung die nach einem der Verfahren der Ansprüche 1 bis 12 codierten Positionscodes decodierbar sind.
- 10 15. Datenübertragungssystem, umfassend eine Vorrichtung nach Anspruch 13 und eine Vorrichtung nach Anspruch 14.

## Zusammenfassung

Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur

5

Die Erfindung betrifft ein Verfahren zur Codierung von Positionen von Datenelementen in einer Datenstruktur, bei dem den Datenelementen Positionscodes in einer vorgegebenen Reihenfolge zugeordnet werden, wobei die Positionscodes derart gewählt werden, dass im Falle von unbegrenzten Codelängen der

10

Positionscodes zwischen den Positionen von zwei Datenelementen beliebig viele weitere Positionscodes zur Codierung von Positionen von weiteren Datenelementen vergeben werden können. Ferner betrifft die Erfindung ein Verfahren zur Codierung

15

von Positionen von Datenelementen in einer Datenstruktur, bei dem den Datenelementen Positionscodes in einer vorgegebenen Reihenfolge zugeordnet werden, wobei die Positionscodes derart gewählt werden, dass zwischen den Positionen von zwei benachbarten Datenelementen weitere Positionscodes zur

20

Codierung von Positionen von weiteren Datenelementen vergeben werden können, wobei die Codelänge von mindestens einem weiteren Positionscodes größer als die längere der Codelängen der Positionscodes der zwei benachbarten Datenelemente ist.

25    Figur 2

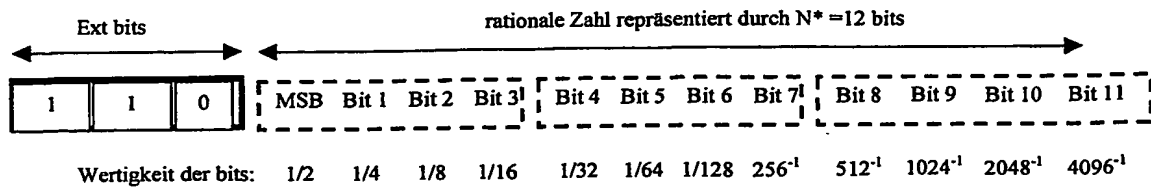


Fig. 1

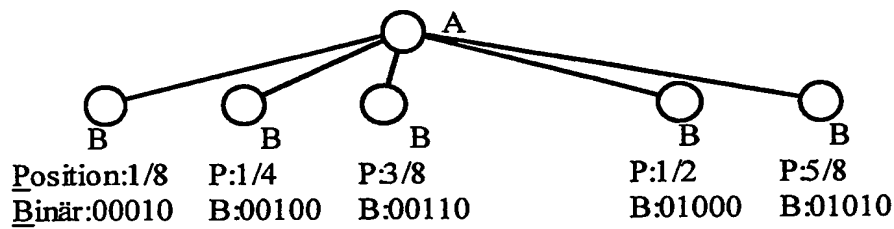


Fig. 2

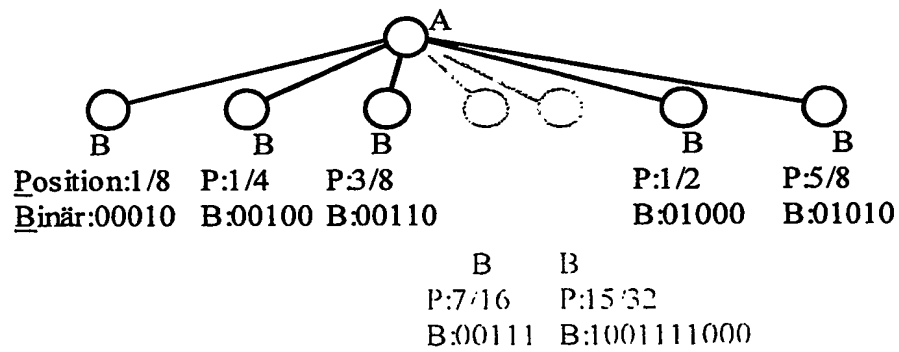


Fig. 3

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☒ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☒ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**